Name: Sahib Bajwa
ID: 107553096

CSCI 3104, Algorithms **Profs. Chen & Grochow**
Problem Set 10 – Due Wed April 22 11:55pm **Spring 2020, CU-Boulder**

Advice 1: For every problem in this class, you must justify your answer: show how you arrived at it and why it is correct. If there are assumptions you need to make along the way, state those clearly.

Advice 2: Informal reasoning is typically insufficient for full credit. Instead, write a logical argument, in the style of a mathematical proof.

Instructions for submitting your solutions:

- All submissions must be typed.
- You should submit your work through the **class Canvas page** only.
- You may not need a full page for your solutions; pagebreaks are there to help Gradescope automatically find where each problem is. Even if you do not attempt every problem, please allot at least as many pages per problem (or subproblem) as are allotted in this template.

Quicklinks: **?? ?? ?? ?? ??**

Name: Sahib Bajwa
ID: 107553096

CSCI 3104, Algorithms | **Profs. Chen & Grochow**
Problem Set 10 – Due Wed April 22 11:55pm | **Spring 2020, CU-Boulder**

1. Consider the following DP table for the Knapsack problem for the list

$$A = [(4, 3), (1, 2), (3, 1), (5, 4), (6, 3)]$$

of (weight, value) pairs. The weight threshold $W = 10$.

- Fill in the values of the table.
- Draw the backward path consisting of backward edges and do not draw (or erase them) the edges that are not part of the optimal backward paths.

(a) Fill the table with the above requirements (You can also re-create this table in excel/sheet).

Weight	Value	I \ w	0	1	2	3	4	5	6	7	8	9	10
4	3	A[0..0]	0	0	0	0	3	3	3	3	3	3	3
1	2	A[0..1]	0	2	2	2	3	3	5	5	5	5	5
3	1	A[0..2]	0	2	2	2	3	5	5	5	6	6	6
5	4	A[0..3]	0	2	2	2	3	5	6	6	6	7	9
6	3	A[0..4]	0	2	2	2	3	5	6	6	6	7	9

(b) Which cell has the optimal value and what is the optimal value for the given problem?

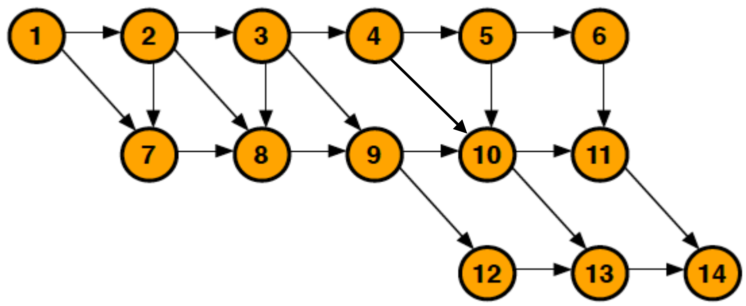
Cell (10, 3) has the optimal value. The optimal value for the given problem is 9.

(c) List out the optimal subset and provide it's weight and value.

The optimal subset for this problem is items: { 0, 1 , 3 }. Its total weight is 10 and the total value is 9. This can be found by going back up the table drawing the backward path. First we select item 3 because item 3's row is the first time the value 9 shows up. Now we have a left over value of 5 (9 - 4 = 5). Next we select item 1 because item 1's row is the first time the value 5 shows up. Now we have a left over value of 3 (5 - 2 = 3). Finally, we select item 0 since item 0's row is the first time value 3 shows up. Now we have a left over value of 0 (3 - 3 = 0), so we have our optimal subset.

Name: Sahib Bajwa
ID: 107553096

2. Given the following directed acyclic graph. Use dynamic programming to fill in a table that counts number of paths from each node j to 14, for $j \geq 1$. Note that a single vertex is considered a path of length 0.



node i	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	1	X(1, 2)	X(1, 3)	X(1, 4)	X(1, 5)	X(1, 2) + 1	X(1, 3) + X(1, 2) + X(1, 7)	X(1, 3) + X(1, 8)	X(1, 4) + X(1, 5) + X(1, 9)	X(1, 6) + X(1, 10)	X(1, 9)	X(1, 10) + X(1, 12)	X(1, 11) + X(1, 13)
2	0	0	1	X(2, 3)	X(2, 4)	X(2, 5)	1	X(2, 7) + 1	X(2, 3) + X(2, 8)	X(2, 4) + X(2, 5) + X(2, 9)	X(2, 6) + X(2, 10)	X(2, 9)	X(2, 10) + X(2, 12)	X(2, 11) + X(2, 13)
3	0	0	0	1	X(3, 4)	X(3, 5)	0	1	X(3, 8) + 1	X(3, 4) + X(3, 5) + X(3, 9)	X(3, 6) + X(3, 10)	X(3, 9)	X(3, 10) + X(3, 12)	X(3, 11) + X(3, 13)
4	0	0	0	0	1	X(4, 5)	0	0	0	X(4, 4) + X(4, 5) + X(4, 9)	X(4, 6) + X(4, 10)	X(4, 9)	X(4, 10) + X(4, 12)	X(4, 11) + X(4, 13)
5	0	0	0	0	0	1	0	0	0	1	X(5, 6) + X(5, 10)	X(5, 9)	X(5, 10) + X(5, 12)	X(5, 11) + X(5, 13)
6	0	0	0	0	0	0	0	0	0	0	1	X(6, 9)	X(6, 10) + X(6, 12)	X(6, 11) + X(6, 13)
7	0	0	0	0	0	0	0	1	X(7, 8)	X(7, 9)	X(7, 10)	X(7, 9)	X(7, 10) + X(7, 12)	X(7, 11) + X(7, 13)
8	0	0	0	0	0	0	0	0	1	X(8, 9)	X(8, 10)	X(8, 9)	X(8, 10) + X(8, 12)	X(8, 11) + X(8, 13)
9	0	0	0	0	0	0	0	0	0	1	X(9, 10)	1	X(9, 10) + X(9, 12)	X(9, 11) + X(9, 13)
10	0	0	0	0	0	0	0	0	0	0	1	X(10, 9)	1	X(10, 11) + X(10, 13)
11	0	0	0	0	0	0	0	0	0	0	0	0	0	1
12	0	0	0	0	0	0	0	0	0	0	0	0	1	X(12, 13)
13	0	0	0	0	0	0	0	0	0	0	0	0	0	1
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0

X(n,m) refers to the number of paths from n to m. Making this table, we can use previously discovered paths to fill the rest of the table. For example. say we want to count the number of paths from 1 to 10. We can use the already discovered number of paths from 1 to 4, 1 to 5, and 1 to 9 and simply add them together (in the table this would be X(1,4) + X(1, 5) + X(1, 9)). This is similar to using already solved sub-problems to solve larger problems.

Note: The "from" nodes are in the column on the left side (blue) and the "to" nodes are in the row at the top (pink-ish).